CHAPTER 1

INTRODUCTION

1.1. Software Development Problems

As the role of software becomes increasingly critical for business as well as for human life, there are three main failures of the software industry namely cost overrun (over budget), schedule overrun (late), and failure to meet requirements specification [CCU87],[CYN96],[EBE95]. Just look at some of the cost overruns at Table 1.1 [WIL98].

Table 1.1 Project Cost Overruns

Project	Minimum Cost Overrun
U.S Office of Mines	$ 15 M
Bank of America	$ 65 M
United Airline Reservation System	$ 145 M
NORAD Update	$ 207 M
U.S. Navy Automated Financial System	$ 446 M
Advanced Logistic System	$ 490 M
U.S. Army TACFIRE	$ 525 M
United Education and Software Inc	$ 650 M
U.S. Army Sgt. York	$ 1,000 M
U.S. Air Force B-1B EW System	$ 1,200 M
British Nimrod	$ 2,200 M

Software is important because it consumes US $ 15 - $ 20 billion per year, employs close to a quarter million highly trained software engineer, is projected to continue a growth trend of about 12% per year, and one of a few high technology industries in which the United States remains preeminent [SHA93].

Loss of life or widespread inconvenience is caused by unreliable software make big headlines in the news media. It is estimated that last few years around 4000 people have died as a result of software defects [SAM98].

In a modern aircraft, if software stops functioning for more than 200 milliseconds, the aircraft is irrecoverable [SAM98]. In June 1996 a European space agency rocket carrying a number of European satellites exploded seconds after its launch. The accident was attributed to software failures [SAM98].

These software failures are caused by the poor of management function that manages software development project [CHI80]. There are 20 problems states in Table 1.2 [CHI80].

As software products become more essential to many number of applications, more attention has been focussed on the areas of software development problems.

Table 1.2 20 Problems In Software Industry

Poor of Management Functions	Problems	Description
Planning	1. Requirements	Requirement specifications are frequently incomplete, ambiguous, inconsistent, and / or un measurable.
	2. Success	Success criteria for software development are frequently inappropriate, which result in "poor quality" delivered software; that is, not maintainable, unreliable, difficult to use, relatively undocumented.
	3. Project	Planning for software engineering project is generally poor.
	4. Cost	The ability to estimate accurately the resources required to accomplish a software development is poor.
	5. Schedule	The ability to estimate accuracy the delivery time on software development is poor.
	6. Design	Decision rules for use in selecting the correct software design techniques, equipment, and aids to be used in designing software engineering project are not available.
	7. Test	Decision rules for use in selecting correct procedure, strategies, and tools to be uses in testing software engineering project are not available.
	8. Maintainability	Procedures, techniques, and strategies for designing maintainable software are not available.
	9. Warranty	Methods to guarantee or warranty that the delivery software will "work" for the user are not available.
	10. Control	Procedures, methods, and techniques for designing a project control system that will enable project managers to successfully control their project are not ready available.
Organizing	11. Type	Decision rules for selecting the proper organizational structure, for example, project, matrix, function, are not available.
	12. Accountability	The accountability structure in many software engineering project is poor, leaving some question as to who is responsible for various project functions.
Staffing	13. Project manager	Procedures and techniques for selecting the correct management techniques for software engineering project management are not available.

Table 1.2 (Continued)

Poor of Management Function	Problems	Description
Controlling	14. Techniques	Decision rules for use in selecting the correct management techniques for software engineering project management are not available.
	15. Visibility	Procedures, techniques, strategies, and aids that will provide visibility of progress(not just resources used) to the project manager are not available.
	16. Reliability	Measurements or indexes of reliability that can be used as an element of software design are not available and there is no way to predict software failure; that is, there is no practical way to show the delivered software meets a given reliability criteria.
	17. Maintainability	Measurements or indexes of maintainability that can be used as an element of software design are not available; that is there is no practical way to show that a given program is more maintainable than another.
	18. Goodness	Measurements or indexes of "goodness" of code that can be used as an element of software design are not available; that is no practical way to show that one program is better than another.
	19. Programmers	Standards and techniques for measuring the quality of performance and the quantity of production expected from programmers and data processing analysis are not available.
	20. Tracing	Techniques and aids that provide an acceptable means of tracing a software development from requirements to completed code are not generally available.

1.2. Software Engineering Concepts

Software Engineering is the application of definition of Engineering to software development. Talking about the engineering, it can be defined as to arrange, manage or carry through by skillful or artful contrivance [JON94]. Software Engineering, the study of disciplines development of

software systems, has reached the point where using traditional techniques cannot make significant breakthrough [TSA90].

Software Engineering is considered as one solution to the software development problems among the other solutions [JON94], [IAN98]. Computer Aided Software Engineering (CASE) is seen by many as people the solution to current quality and productivity problems. Many have suggested that CASE technology has the potential for revolutionize the way software that is built by transforming current processes of "**Crafting**" software into an Engineering discipline [VIP90].

1.3. Standard Software Process Improvement

The objectives of CASE system are to improve the productivity during the software development progress and the quality of software using Software Engineering Concepts via automation of the software development life cycle.

The industry has realized that CASE tools are not enough. One fact that the software industry has been established is that " **a fool with a tool is still a fool** "[SAM98].

The experience of software industry with CASE tools has proved that the main reason for failing software projects has a little to do with technology and tools, and much to do with lack of process disciplines. Only by creating a disciplined process for software development we can manage and control the quality of software product.

Because of that, it is needed a practical approach for setting up a disciplined and continuously improving software process environment. Currently, there are a number of software process improvement models and emerging standards developed by international organizations, industry consortia, large software purchaser and software developers. So far, the most popular model for software process improvement are Capability Maturity Model (CMM) and ISO 9001 with its associated guide ISO 9000-3 [AJA97],[SAM98],[ISO96].

1.4. Total Quality Management Concepts

ISO 9000 is focused on technical system. Technical capability is no longer the principle competitive determinant in computer and software industry. Technical capability is necessary but not sufficient for success.

Software organization are more than technical system. A part from anything else, they are also social system. They are about people and about the way people behave and interact with each other in groups. They are about the attitude, the aspiration, and the motivation of people in work situation.

The technical system needs to be integrated with social system in order to build a quality culture. Therefore, Total Quality Management is the integrating of these systems through the adoption of managerial process which provide a focus on customer needs, employee needs, and the needs of stakeholders of the organization.

According to Arthur (1992) the problem with software is not quality, **Quality is the solution to the problem** [ART92].

1.5. Approach To Overcome Software Development Problems

To overcome the software development problems is essential to ensure the reliable product and services, and to gain the customers satisfaction.

This thesis deals with the current software problems and application of the new quality oriented methods, as characterized by Total Quality Management, Quality System Standard ISO 9001 and SEI - CMM levels.

Total Quality Management makes quality a way of totally focusing the organization on the competitive discipline of serving the customer. In general, this standard is being adopted by world wide in industry in general and a software derivative has been developed (ISO 9000 - 3).

The integration of the Total Quality Management, Software Engineering concept, and the Quality System Standard ISO 9000 - 3 can be applied by proposing " **A Model of Compliance On The Quality System for Software Industry By ISO 9000 - 3"** [ZAR99].

1.6. Objectives And Benefit of Project

The main objectives of the project are :

1. To establish a model of self assessment tool to evaluate the current quality level of software organization . The result of evaluation is used

as the basic to improve the process quality of software development continuously and to prevent the non conformance on the every stages from design, production, and services.

2. To prepare the arrangement steps of quality system with standard on ISO 9000-3 as well as to make preparation to get certification of ISO 9001.
3. To establish a Total Quality Management system for the software development and maintenance.

An extensive survey of benefit of ISO certification or registration has been conducted by Lai [LAI95]. The results are summarized in Table 1.3. The survey shows that companies have benefited most from improvement in quality consciousness, improvement in existing quality system and in the documentation of quality system.

Table 1.3 Benefits obtained For Achieving ISO 9000 Certification

Factor	Benefits (Mean)
Securing New /Present Market	
1. Secure export opportunities to European market	3.08
2. Secure export opportunities to other market	2.93
Improve Competitive Position	
3. Improve Competitive Position	3.48
Customer Pressure and Satisfaction	
4. Improve customers' confidence	3.65
5. Comply with customers' requirements	3.34
6. Better customers' satisfaction	3.55
Practical Benefits	
7. Improve quality consciousness	3.66
8. Improve existing quality system	3.88

Table 1.3 (Continued)

9. Documentation of existing quality system	3.86
10. Reduce quality system	3.42
11. Improve process	3.31
12. Help to attain TQM	3.27
13. Achieve better team work and cooperation among employees	3.34
14. Reduce rework and wastage	3.30
15. Improve communication among employee	3.42
16. Improve productivity	2.95
17. Reduce cost	2.85
18. Improve morale of employees	2.92
19. Reduction in supplier quality audits by customers	2.97
20. Increase in profits.	2.77

5 - Critically Important 4 - Very Important 3 - Good to have
2 - Not Important 1 - Disagree

1.7. Methodology

Methodology used covers reading the literature in the library, getting the information through Internet(Website), having comparative study in manufacturing organizations which has been successful in implementing ISO 9001, and making a survey in the software organization , design the model of project at FSKTM lab.

The activities can be described in detail as follow :

1. Literature study and collecting materials in the matter of software engineering, software process improvement, Total Quality Management, Quality system standard ISO 9000-3 from library and Internet.
2. Design a list of questions of assessment to the existing quality system for the software industry to obtain the quality information.

3. Analysis and design the module of application under windows operating system and using the Delphi Language (Visual Pascal).
4. The output expected from the application are: management compliance summary, assessment result summary and the weaknesses report of software industry that need improvement and step to do the improvement.

1.8. Contents of The Thesis

The thesis is structured in eight chapters followed by a list of references, Figures, Tables and Appendix.

Chapter One describes software development problems, software engineering concepts, total quality management concepts, standard software process improvement, approach to overcome the development problems, objective and benefit of project and methodology of the study.

Chapter Two describes the definition of software, software engineering, software quality, principles of software engineering, tools and techniques that are often used for solving the problems in software development activities.

Chapter Three describes and presents ISO 9000-3 standard, Capability Maturity Model, and relationship between ISO 9001, ISO 9000-3 and CMM.

Chapter Four describes the definition of quality term, Total Quality Management (TQM) , TQM Principles, relationship between TQM and

ISO 9000, implementation of TQM in software industry, the role of software engineering in TQM.

Chapter Five focuses on the requirement analysis of ISAT903 which encompasses the analysis and the solution of software development problems, standard for software process assessment and requirements specification.

Chapter Six focuses on the design of ISAT903 which encompasses approach used, tools and techniques, data flow diagrams of ISAT903, and structure charts of ISAT903.

Chapter Seven highlights the implementation of ISAT903. The algorithms for the various functional modules of ISAT903 are formulated and described in detail. The approaches and strategies of testing users performance of ISAT903, features and characteristics of ISAT903, application and limitation of ISAT903 are also described in this chapter.

Chapter Eight describes conclusion and suggestions.